作业2

苏州大学 计算机科学与技术学院
汪笑宇
Email: xywang21@suda.edu.cn

作业2-1

■4.4-2 $T(n)=T(n/2)+n^2$

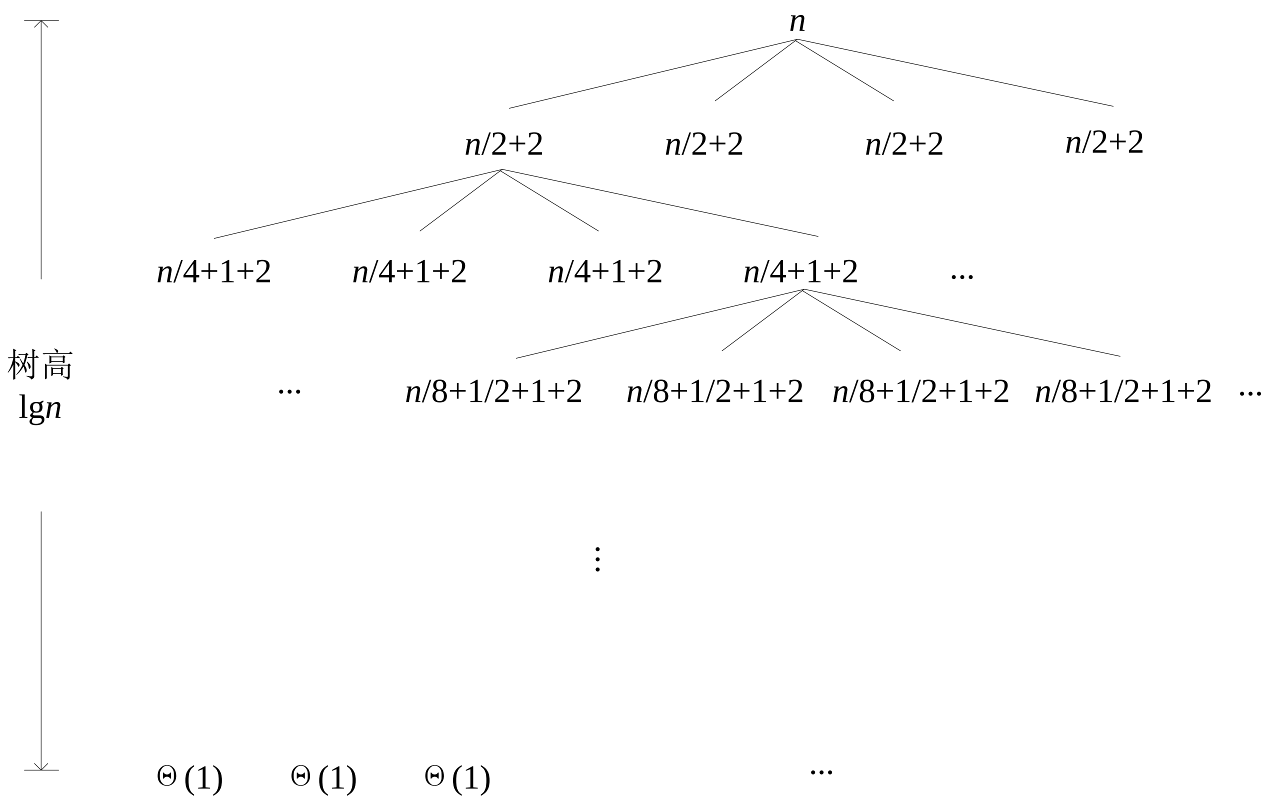
树高		代价
	n^2	n^2
	$n^2/4$	$n^2/4$
$\lg n$	$n^2/16$	$n^2/16$
	$\vdots$	$\vdots$
		$n^2/(2^{\lg n-1})^2=4$
	$\Theta(1)$	$\Theta(1)$

$$
\begin{aligned}
T(n) &= n^2 + \frac{n^2}{4} + \frac{n^2}{4^2} + \ldots + \frac{n^2}{4^{\lg n-1}} + \Theta(1) \\
&= n^2 \cdot \frac{1-(1/4)^{\lg n}}{1-1/4} + \Theta(1) \\
&< \frac{4}{3}n^2 + \Theta(1) = O(n^2)
\end{aligned}
$$

代入法验证：
即需证明可恰当选择常数$c>0$，使得$T(n)\le cn^2$
假设对于所有的$m<n$都有$T(m)\le cm^2$，特别的，对于$m=n/2$有$T(n/2)\le cn^2/4$，则
$T(n)\le cn^2/4+n^2=(c/4+1)n^2\le cn^2$
当$c\ge 4/3$时成立
取$c=4/3$，边界条件$T(1)=1\le 4/3$满足条件
因此$T(n)=O(n^2)$成立

作业2-1 (续)

■4.4-3 $T(n)=4T(n/2+2)+n$



代价

n

$4(n/2+2)$

$4^2(n/4+1+2)$

$4^3(n/8+1/2+1+2)$

$\vdots$

$$4^{\lg n-1}\left(n/2^{\lg n-1}+\sum_{i=0}^{\lg n-2}2\times(1/2)^i\right)$$

$$=4^{\lg n-1}\left(n/2^{\lg n-1}+\frac{2-2\times(1/2)^{\lg n-1}}{1-1/2}\right)$$

$$=4^{\lg n-1}\left(n/2^{\lg n-1}+4-4/2^{\lg n-1}\right)$$

$\Theta(4^{\lg n})=\Theta(n^2)$

n = 1,2,3,4时都是递归边界条件，否则递归无法终止或导致矛盾

假设：$T(1) = T(2) = T(3) = T(4) = 1$

作业2-1 (续)

■4.4-3 $T(n)=4T(n/2+2)+n$

$$
\begin{aligned}
T(n) &= n + 4(n/2+2) + \ldots + 4^{\lg n-1}(n/2^{\lg n-1} + 4 - 4/2^{\lg n-1}) + \Theta(n^2) \\
&= \sum_{i=0}^{\lg n-1} 4^i((n-4)/2^i + 4) + \Theta(n^2) \\
&= \sum_{i=0}^{\lg n-1} 2^i n - \sum_{i=0}^{\lg n-1} 2^{i+2} + \sum_{i=0}^{\lg n-1} 4^{i+1} + \Theta(n^2) \\
&= n \cdot \frac{1-2^{\lg n}}{1-2} - \frac{4-2^{\lg n+2}}{1-2} + \frac{4-4^{\lg n+1}}{1-4} + \Theta(n^2) \\
&= n(n-1) - (4n-4) + (4n^2-4)/3 + \Theta(n^2) \\
&= O(n^2)
\end{aligned}
$$

作业2-1 (续)

■4.4-3 $T(n)=4T(n/2+2)+n$

代入法验证：（加入修正项）

证明可恰当选择常数$c,d>0$，使得$T(n)\le c(n-4)^2-d(n-4)$

假设对于所有的$m<n$都有$T(m)\le c(m-4)^2-d(m-4)$，特别的对于$m=n/2+2<n$，有

$T(n/2+2) \le c(n/2+2-4)^2-d(n/2+2-4) = c(n/2-2)^2-d(n/2-2)$，其中$n\ge 5$

则

$T(n) \le 4(c(n/2-2)^2-d(n/2-2))+n = c((2(n/2-2))^2)-2d(2(n/2-2))+n= c(n-4)^2-2d(n-4)+n$

$\le c(n-4)^2-d(n-4)$

当c为任意正数且$-2d(n-4)+n\le -d(n-4)$即$d\ge 1/(1-4/n)$时成立。因$n\ge 5$，则$d\ge 5$。

因$T(5)= 4T(4)+5=9 \le c-d$，

取$c=14$, $d=5$, $n_0=5$，边界条件$T(5) = 9 \le 14*(5-4)^2-5*(5-4)=9$满足条件

因此$T(n) = O(n^2)$成立

作业2-1 (续)

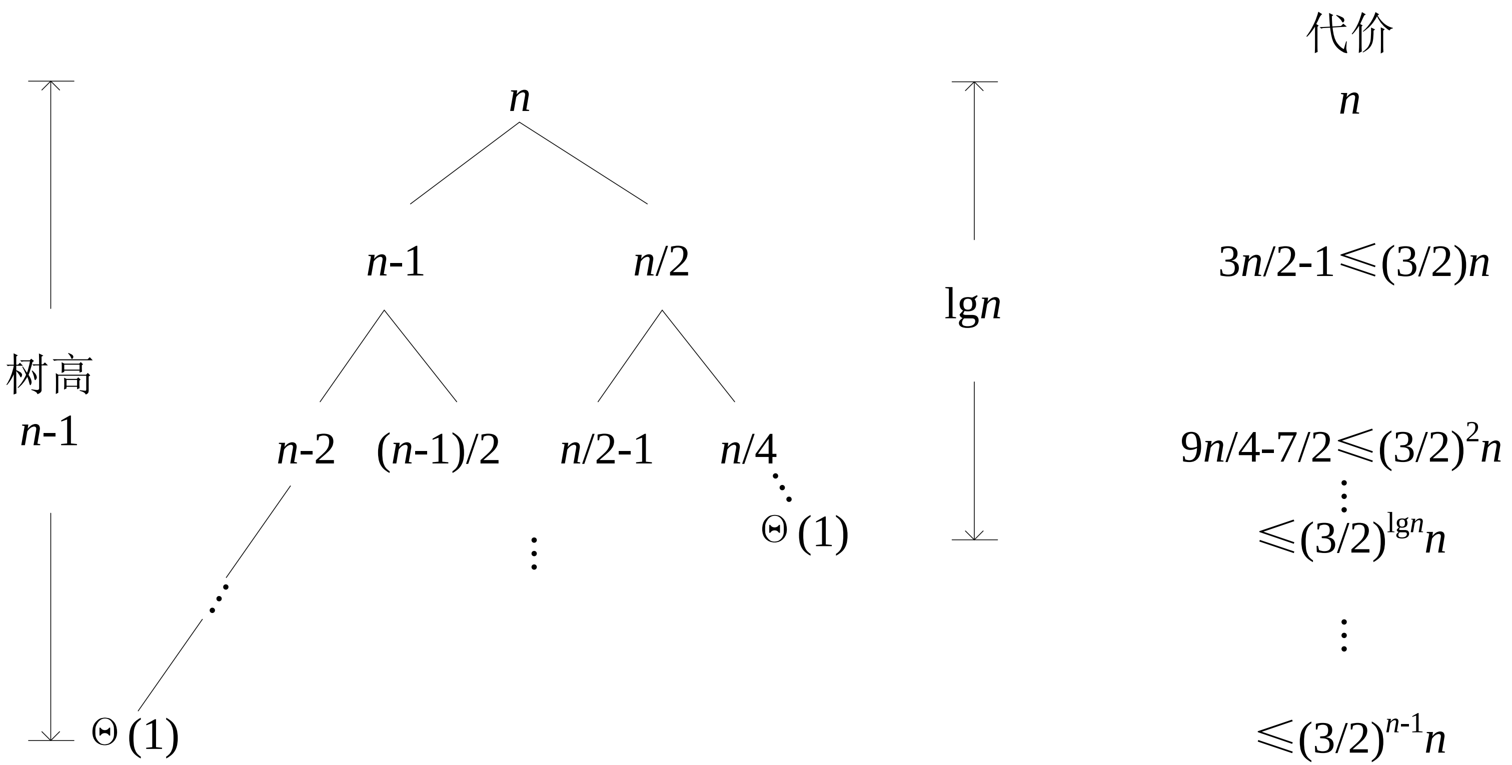
■4.4-4 $T(n)=T(n-1)+1$

$$T(n) = T(n-1) + 1 = T(n-2) + 2 = T(n-3) + 3$$
$$= \ldots = T(n-(n-1)) + (n-1) = O(n)$$

代入法验证：
即需证明可恰当选择常数$c>0$，使得$T(n)\le cn$
假设对于所有的$m<n$都有$T(m)\le cm$，
特别的，对于$m=n-1$ 有$T(n-1) \le c(n-1)$，则
$T(n) \le c(n-1)+1 = cn-(c-1) \le cn$
当$c\ge 1$时成立
取$c=1$，边界条件$T(1) = 1 \le 1$满足条件
因此$T(n) = O(n)$成立

作业2-1 (续)

■4.4-5 $T(n)=T(n-1)+T(n/2)+n$



作业2-1 (续)

■4.4-5 $T(n)=T(n-1)+T(n/2)+n$

$$T(n) < \sum_{i=0}^{n-1} (3/2)^i n$$

$$= n \cdot \frac{1-(3/2)^n}{1-3/2} = 2n \cdot (3/2)^n - 2n = O(2^n)$$

代入法验证：
即需证明可恰当选择常数$c>0$，使得$T(n) \le c2^n$
假设对于所有的$m<n$都有$T(m) \le c2^m$，则
$T(n) \le c2^{n-1}+c2^{n/2}+n = c/2 \cdot 2^n + c(\sqrt{2})^n + n \le c2^n$
当$n \ge 3$且$c \ge \frac{3}{4-2\sqrt{2}} \approx 2.56$ 时成立
取$c=3$，边界条件$T(1) = 1 \le 6$, $T(2) = 2T(1)+2 = 4 \le 12$
满足条件
因此$T(n) = O(2^n)$成立

作业2-2 a-c

4-1 （递归式例子） 对下列每个递归式，给出 $T(n)$的渐近上界和下界。假定 $n \leqslant 2$ 时 $T(n)$是常数。给出尽量紧确的界，并验证其正确性。

a. $T(n)=2T(n/2)+n^4$

b. $T(n)=T(7n/10)+n$

c. $T(n)=16T(n/4)+n^2$

d. $T(n)=7T(n/3)+n^2$

e. $T(n)=7T(n/2)+n^2$

f. $T(n)=2T(n/4)+\sqrt{n}$

依照递归式形式$T(n) = aT(n/b) + f(n)$进行说明：

a. $a=2, b=2, n^{\log_b a}=n^{\lg 2}=n, f(n)=n^4=\Omega(n^4)=\Omega(n^{1+\epsilon})$
其中ε=4–1=3>0。当c=1/8且n足够大时，$af(n/b) = 2(n/2)^4=n^4/8 \leq n^4/8 = cf(n)$
根据主定理第三种情况，$T(n) = \Theta(n^4)$

b. $a=1, b=10/7, n^{\log_b a}=n^{\log_{\frac{10}{7}} 1}=1, f(n)=n=\Omega(n)=\Omega(n^{0+\epsilon})$
其中ε=1–0=1>0。当c=7/10且n足够大时，$af(n/b) = 7n/10 \leq 7n/10 = cf(n)$
根据主定理第三种情况，$T(n) = \Theta(n)$

c. $a=16, b=4, n^{\log_b a}=n^{\log_4 16}=n^2, f(n)=n^2$，根据主定理第二种情况，
$T(n) = \Theta(n^2\lg n)$

作业2-2 d-f

4-1 （递归式例子） 对下列每个递归式，给出 $T(n)$的渐近上界和下界。假定 $n\leqslant 2$ 时 $T(n)$是常数。给出尽量紧确的界，并验证其正确性。

a. $T(n)=2T(n/2)+n^4$

b. $T(n)=T(7n/10)+n$

c. $T(n)=16T(n/4)+n^2$

d. $T(n)=7T(n/3)+n^2$

e. $T(n)=7T(n/2)+n^2$

f. $T(n)=2T(n/4)+\sqrt{n}$

依照递归式形式$T(n) = aT(n/b) + f(n)$进行说明：

d. $a=7, b=3, n^{\log_b a}=n^{\log_3 7}, f(n)=n^2=\Omega(n^2)=\Omega(n^{\log_3 7+\epsilon})$

其中ε=2–$\log_3 7$>0。当c=7/9且n足够大时，$af(n/b) = 7(n/3)^2=7n^2/9 \leq 7n^2/9 = cf(n)$

根据主定理第三种情况，$T(n) = \Theta(n^2)$

e. $a=7, b=2, n^{\log_b a}=n^{\lg 7}, f(n)=n^2=O(n^{\lg 7})=O(n^{2+\epsilon})$

其中ε=lg7–2>0。根据主定理第一种情况，$T(n) = \Theta(n^{\lg 7})$

f. $a=2, b=4, n^{\log_b a}=n^{\log_4 2}=n^{1/2}, f(n)=n^{1/2}$，根据主定理第二种情况，

$T(n)=\Theta(\sqrt{n}\lg n)$

作业2-3

- 受限汉诺塔问题：有3个塔座从左到右分别为X、Y、Z，每次只能将一个塔座上最小的圆盘移动到相邻塔座上，即不可将圆盘从X直接移动到Z（或者从Z 直接移动到X）。当然，依旧不允许大圆盘放在小圆盘上。任务为：将n个大小不同且依次叠放的圆盘从X 移动到Z。
- 请设计一个递归算法：(1) 说明算法设计思想，(2) 写出伪代码，(3) 分析使用该算法需移动圆盘的次数。

作业2-3 (续)

- 设计思想：每次只能移动到相邻塔座，因此最下方圆盘需要先从X移动到Y，再从Y移动到Z。由此，上方$n-1$个圆盘要先移动到Z，待最下方圆盘移走后再移回X，最下方圆盘移动到Z后，上方圆盘再移动到Z。
  - 分解：设$n \geq 1$，将n个圆盘从X移到Z，Y为辅助塔座：
    - 将上面n-1个盘从X移至Z，Y为辅助塔座
    - 将第n号圆盘（最大的圆盘）从X移至Y
    - 将Z上n-1个圆盘移至X，Y为辅助塔座
    - 将第n号圆盘（最大的圆盘）从Y移至Z
    - 将X上n-1个圆盘移至Z，Y为辅助塔座
  - 终结条件：n=0时，不需要任何操作

作业2-3 (续)

```
RESTRICTED_HANOI(n, X, Y, Z)
if n = 0
        return
else RESTRICTED_HANOI(n – 1, X, Y, Z)  //上方n – 1个圆盘移动到Z
        move(n, X, Y)  //n号圆盘移动到Y
        RESTRICTED_HANOI(n – 1, Z, Y, X)   //上方n – 1个圆盘移动到X
        move(n, Y, Z)   //n号圆盘移动到Z
        RESTRICTED_HANOI(n – 1, X, Y, Z)   //上方n – 1个圆盘移动到Z
```

作业2-3 (续)

■移动次数：$T(n) = 3T(n-1) + 2$，其中$T(0) = 0$

$$
\begin{aligned}
T(n) &= 3T(n-1) + 2 \\
&= 3(3T(n-2) + 2) + 2 \\
&= 3(3(3T(n-3) + 2) + 2) + 2 \\
&\;\vdots \\
&= 3^n T(n-n) + 2 \cdot (3^0 + 3^1 + \ldots + 3^{n-1}) \\
&= 3^n - 1 \\
&= O(3^n)
\end{aligned}
$$

作业2-4

2-4 （逆序对） 假设 $A[1..n]$ 是一个有 n 个不同数的数组。若 $i<j$ 且 $A[i]>A[j]$，则对偶 (i, j) 称为 A 的一个**逆序对**（inversion）。

a. 列出数组〈2，3，8，6，1〉的 5 个逆序对。

b. 由集合{1，2，…，n}中的元素构成的什么数组具有最多的逆序对？它有多少逆序对？

c. 插入排序的运行时间与输入数组中逆序对的数量之间是什么关系？证明你的回答。

d. 给出一个确定在 n 个元素的任何排列中逆序对数量的算法，最坏情况需要 $\Theta(n\lg n)$ 时间。（提示：修改归并排序。）

a. (1, 5), (2, 5), (3, 5), (4, 5), (3, 4)
b. 从大到小排序的数组<n, n-1, n-2, ..., 2, 1>具有最多逆序对，任两个元素都构成逆序对，因此总共有 $C_n^2 = n(n-1)/2$ 个逆序对
c. 插入排序运行时间与输入数组中逆序对数量呈线性关系

作业2-4 (续)

INSERTION_SORT(*A*)	代价	次数
1 **for** $j \leftarrow 2$ **to** *A.length* **do**	c_1	n
2 $\quad key \leftarrow A[j]$	c_2	n-1
3 $\quad$ // Insert $A[j]$ into the sorted sequence $A[1..j\text{-}1]$	0	n-1
4 $\quad i \leftarrow j - 1$	c_4	n-1
5 $\quad$ **while** $i>0$ and $\boxed{A[i]>key}$ **do**	c_5	$\sum_{j=2}^{n} t_j$
6 $\qquad A[i+1] \leftarrow A[i]$	c_6	$\sum_{j=2}^{n}(t_j-1)$
7 $\qquad i \leftarrow i - 1$	c_7	$\sum_{j=2}^{n}(t_j-1)$
8 $\quad A[i+1] \leftarrow key$	c_8	n-1

t_j：对于值j，第5行执行while循环测试的次数

$A[i]>key$满足一次条件则存在一个逆序对(i, j)，因此while循环的循环体执行一次则存在一个逆序对，即$I(n)=\sum_{j=2}^{n}(t_j-1)$

由此可写出总代价$T(n)=aI(n)+f(n)$，其中a是一个常数，且当n确定时$f(n)$也是一个常数，因此与逆序对数量关系呈线性关系

作业2-4 (续)

d. 基本思想：归并排序在合并阶段可对逆序对进行统计，若右半子数组元素$A[j]$较左半子数组元素$A[i]$先加入最终序列，则说明当前元素$A[j]$与左半子数组剩下的所有元素都构成逆序对，此时逆序对数量加上左半子数组剩下元素数量$n_1 - i + 1$

```
MERGE_INVERSION(A, p, q, r)
n1 ← q – p + 1
n2 ← r – q
let L[1..n1+1] and R[1..n2+1] be new arrays
for i ← 1 to n1 do L[i] ← A[p+i–1]
for j ← 1 to n2 do R[j] ← A[q+j]
L[n1+1] ← ∞,  R[n2+1] ← ∞
i ← 1,  j ← 1, inv ← 0
for k ← p to r do
    if L[i] ≤ R[j]
        A[k] ← L[i],  i ← i + 1
    else A[k] ← R[j],  j ← j + 1, inv ← inv + n1 – i + 1
return inv
```

```
INVERSION(A, p, r)
if p < r
    q ← ⌊(p + r)/2⌋
    left ← INVERSION(A, p, q)
    right ← INVERSION(A, q+1, r)
    return left + right + MERGE_ INVERSION(A, p, q, r)
```